第六章 功率谱估计

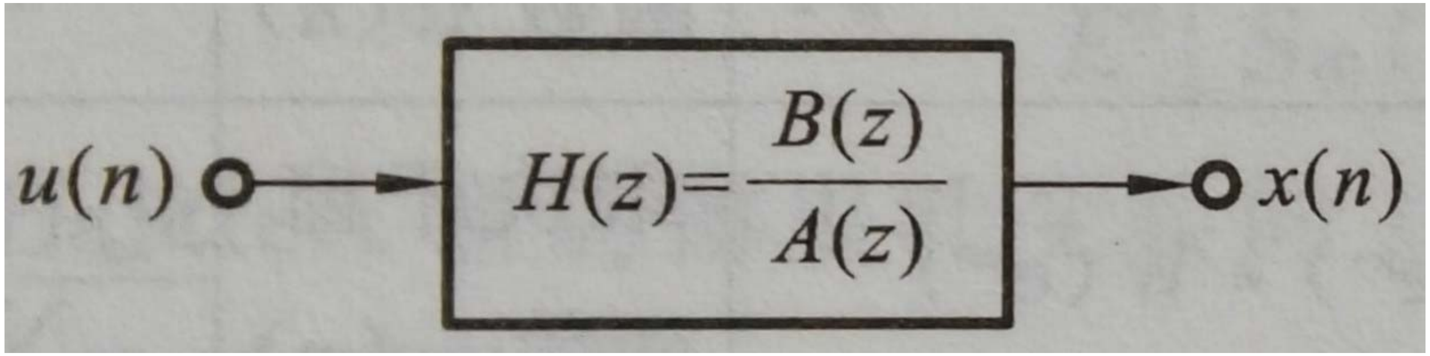
§3.4 随机过程的参数模型

3.4.1 概述

- 谱估计的现代方法，用参数模型描述随机过程的2阶矩，因此，称为参数模型法，简称模型法或参数法。

- 使用参数方法计算功率谱，依据的是功率谱是模型参数的函数，而不是直接依据功率谱与自相关序列的函数关系。

- 广义平稳随机过程理论和实验结果表明：实际应用中遇到的**大多数离散时间随机过程适合用下图所示的有理传输函数模型描述**。



- 输入 $u(n)$：零均值、方差为 σ^2 的白噪声；
- $H(z)$：有理传输函数的线性时不变系统；
- 输出 $x(n)$：被模拟的离散时间随机信号；

- 由于输入激励是白噪声，其频谱为平行于x轴的一条直线，所以模型输出信号的功率谱的形状与线性是不变系统的振幅特性的平方相同，模型输出的功率谱完全由白噪声的方差和线性是不变系统的参数确定。

- 用参数方法进行谱估计，一般包含三个步骤：

1. 根据掌握的关于随机过程的先验知识选择合适的模型；
2. 由已知的随机过程的有限个观测数据估计自相关函数，并有自相关函数计算模型参数；
3. 利用估计的模型参数计算功率谱。

- 用参数模型方法估计的功率谱，不需要对数据进行加窗截断，因而比用周期图有更好的性能，主要是有更高的分辨率；
- 性能改善的程度，通常取决于：
  - ✓ 选择的模型对被估计随机过程逼近的程度；
  - ✓ 选择的模型对已知观测数据或自相关函数的拟合程度。

- 有理传输函数模型包括三种：
  - ✓ 自回归模型(Autoregressive, AR)；
  - ✓ 滑动平均模型(Moving Average, MA)；
  - ✓ 自回归滑动平均模型(Autoregressive-Moving, Average, ARMA)。

- 自回归模型(AR)：

  适用于描述有尖峰但无深谷的功率谱；

- 滑动平均模型(MA)：

  适用于有深谷、无尖峰的功率谱；

- 自回归滑动平均模型(ARMA)：

  适用于既有尖峰也有深谷的模型。

对于快速衰减的功率谱，三种模型都不很准确。

- 自回归模型(AR)：也适用描述具有随机相位的正弦信号的功率谱，因为正弦信号可看成是频带很窄的随机信号，因此，其自相关函数可以由参数很少的AR过程来得到；

- 具体而言，M个实正弦信号，可以用一个$2M$阶AR模型描述；M个复正弦信号，只需用一个M阶AR模型描述。

◆三种模型中，AR模型在谱估计中获得了最广泛的应用，主要原因是:

✓描述AR模型参数与自相关函数关系的Yule-Walker方程是一组线性方程，而MA或ARMA模型的Yule-Walker方程则是高度非线性的；

✓只要选择的AR模型适用于被估计的随机过程，由AR模型得到的功率谱估计，与经典谱估计相比，其偏差和方差都更小。

3.4.2 离散时间随机信号的有理传输函数

- 对一个有理传输函数，其输入激励$u(n)$与输出$x(n)$之间的关系用常系数线性差分方程描述：

$$x(n)=-\sum_{k=1}^{p}a_k x(n-k)+\sum_{k=0}^{q}b_k u(n-k) \qquad (3.4.1)$$

或

$$x(n)=\sum_{k=0}^{\infty}h(k)u(n-k) \qquad (3.4.2)$$

其中，是因果线性时不变系统的冲激响应。

其模型框图如下所示：

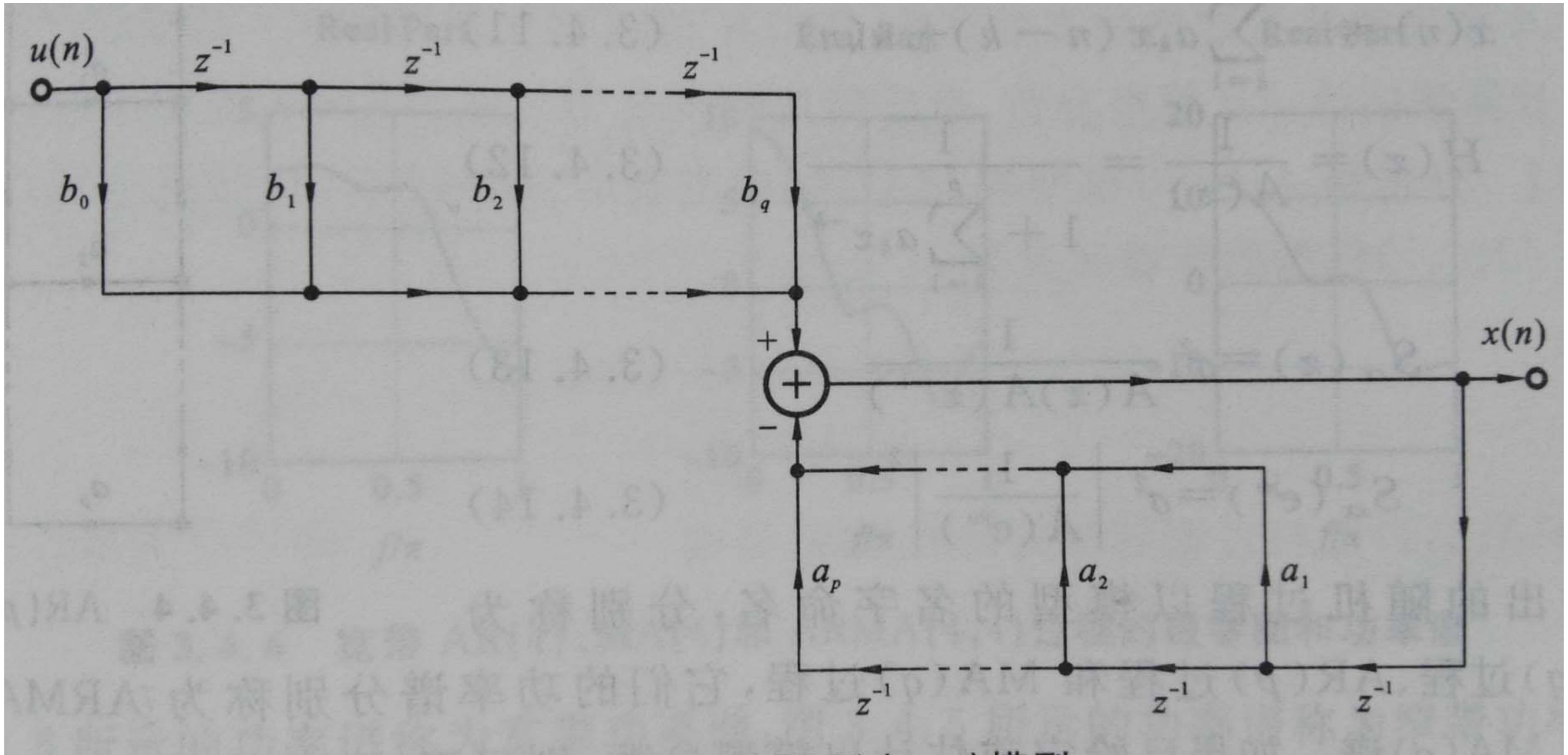


图 3.4.2　ARMA(p,q)模型

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + \sum_{k=0}^{q} b_k u(n-k)$$

- 差分方程$x(n)=-\sum_{k=1}^{p}a_k x(n-k)+\sum_{k=0}^{q}b_k u(n-k)$

对应的线性时不变系统的有理传输函数为

$$H(z)=\frac{B(z)}{A(z)}=\frac{\sum_{k=0}^{q}b_k z^{-k}}{1+\sum_{k=1}^{p}a_k z^{-k}} \qquad (3.4.3)$$

其中,b_k 是系统前馈支路的系数，称为滑动平均或MA系数；a_k是系统反馈支路的系数，称为自回归或AR系数。

- 假设 $H(z)$ 的全部极点都在单位圆内，使系统是稳定、因果的，因此输出是广义平稳的。

□ 由于输入 $u(n)$ 是均值为零、方差为 σ^2 的白噪声，所以输出的功率谱为

$$S_{xx}(z)=\sigma^2 H(z)H^*(1/z^*)=\sigma^2\frac{B(z)B^*(1/z^*)}{A(z)A^*(1/z^*)} \quad (3.4.4)$$

如果 $h(n)$ 是实数，有 $H^*(1/z^*)=H(z^{-1})$，上式化为

$$S_{xx}(z)=\sigma^2 H(z)H(z^{-1})=\sigma^2\frac{B(z)B(z^{-1})}{A(z)A(z^{-1})} \quad (3.4.5)$$

□因此，用数字频率表示的功率谱为

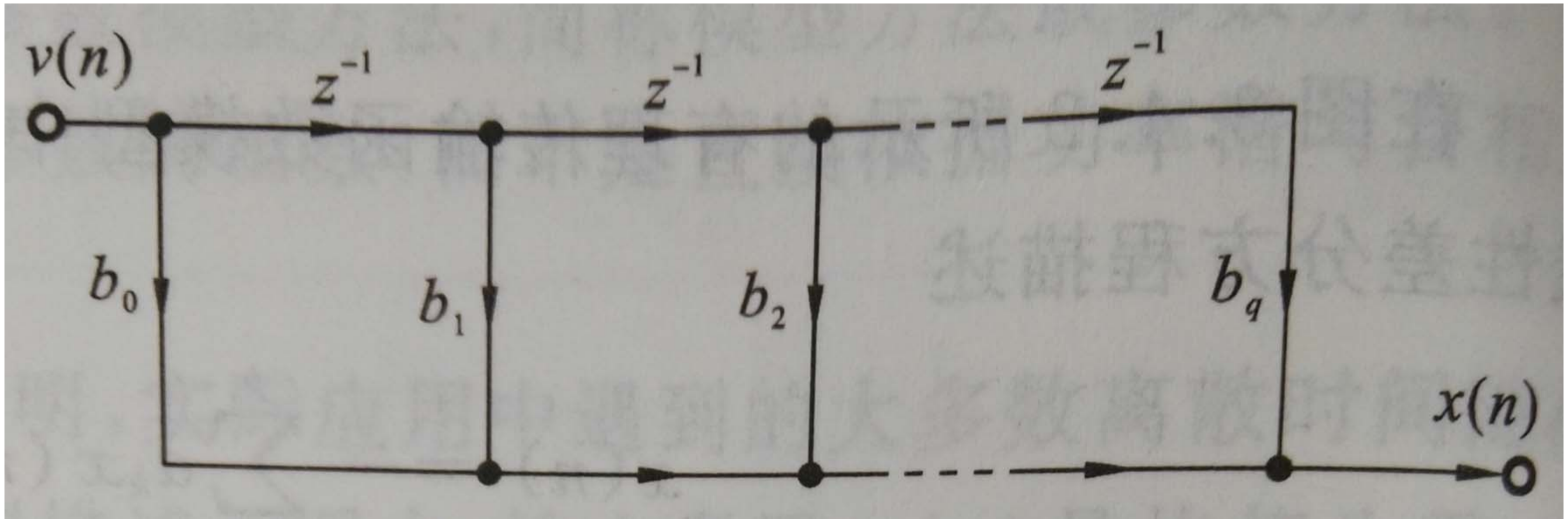
$$S_{xx}(e^{jw}) = \sigma^2 \left|H(e^{jw})\right|^2 = \sigma^2 \left|\frac{B(e^{jw})}{A(e^{jw})}\right|^2 \qquad (3.4.6)$$

根据观测数据估计出了模型参数 a_k 和 b_k 以及功率 σ^2，将它们带入上式，即可计算出功率谱的估计值。$H(z)$ 的增益可归并入一起考虑，因此，可假设 $b_0=1$。

上述一般情况的有理传输函数模型称为(*p*,*q*)阶自回归滑动平均模型或ARMA(*p*,*q*)模型，也称为(*p*,*q*)阶零点-极点模型。

➢ 如果除 $a_0=1$，所有的 $a_k=0$，则得到q阶滑动平均模型或MA(q)模型，也称为q阶全零点模型，即

$$x(n)=\sum_{k=0}^{q} b_k u(n-k) \qquad (3.4.7)$$



将式(3.4.7) $x(n)=\sum_{k=0}^{q} b_k u(n-k)$ 进行z变换，可得传递函数

$$H(z)=B(z)=\sum_{k=0}^{q} b_k z^{-k} \qquad (3.4.8)$$

则输出 $x(n)$ 的功率谱为

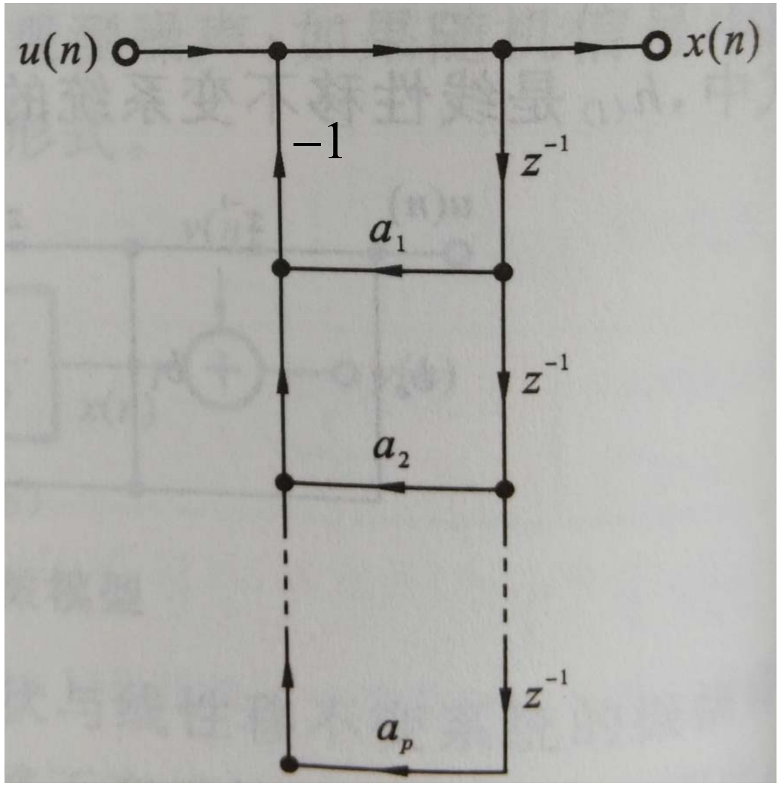
$$S_{xx}(z)=\sigma^2 B(z)B(z^{-1}) \qquad (3.4.9)$$

用数字频率表示的功率谱为

$$S_{xx}(e^{jw})=\sigma^2 \left|B(e^{jw})\right|^2 \qquad (3.4.10)$$

- 如果除 $b_0=1$ ，所有的$b_k=0$ ，则得到p阶自回归模型或AR(p)模型，也称为p阶全极点模型，即

$$x(n)=-\sum_{k=1}^{p}a_k x(n-k)+u(n) \qquad (3.4.11)$$



将式(3.4.11) $x(n) = -\sum_{k=1}^{p} a_k x(n-k) + u(n)$ 进行z变换，可得传递函数

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{k=1}^{p} a_k z^{-k}} \qquad (3.4.12)$$

则输出 $x(n)$ 的功率谱为

$$S_{xx}(z) = \sigma^2 \frac{1}{A(z)A(z^{-1})} \qquad (3.4.13)$$

用数字频率表示的功率谱为

$$S_{xx}(e^{jw}) = \sigma^2 \left| \frac{1}{A(e^{jw})} \right|^2 \qquad (3.4.14)$$

模型输出的随机过程，以模型的名字命名，分别称为：

- ✓ ARMA(p,q)过程
- ✓ AR (p)过程
- ✓ MA(q)过程

它们的功率谱分别称为：

- ✓ ARMA(p,q)谱
- ✓ AR (p)谱
- ✓ MA(q)谱

如果给定或估计出模型参数，则可用式

$$S_{xx}(e^{jw}) = \sigma^2 \left|H(e^{jw})\right|^2 = \sigma^2 \left|\frac{B(e^{jw})}{A(e^{jw})}\right|^2 \qquad (3.4.6)$$

$$S_{xx}(e^{jw}) = \sigma^2 \left|B(e^{jw})\right|^2 \qquad (3.4.10)$$

$$S_{xx}(e^{jw}) = \sigma^2 \left|\frac{1}{A(e^{jw})}\right|^2 \qquad (3.4.14)$$

分别计算出三种模型的随机过程的功率谱。

3.4.3 三种模型参数之间的关系

- 一个有限阶的ARMA过程或AR过程，可以用唯一的无限阶的MA模型表示；
- 一个有限阶的ARMA过程或MA过程，可以用唯一的无限阶的AR模型表示；
- AR模型通常有成熟的高效算法，且其建立的方程是线性的，所以在利用MA或ARMA进行谱估计时，总是先使用高效算法估计AR模型参数，然后再转换成MA或ARMA模型。

3.4.4 Yule-Walker方程

- ARMA(p,q)模型的Yule-Walker方程

➢ 将式(3.4.1) $x(n)=-\sum_{k=1}^{p}a_k x(n-k)+\sum_{k=0}^{q}b_k u(n-k)$

两端同乘以 $x^*(n-m)$ 后取期望，可得

$$E[x(n)x^*(n-m)]=-\sum_{k=1}^{p}a_k E[x(n-k)x^*(n-m)]$$

$$+\sum_{k=0}^{q}b_k E[u(n-k)x^*(n-m)]$$

即

$$R_{xx}(m)=-\sum_{k=1}^{p}a_k R_{xx}(m-k)+\sum_{k=0}^{q}b_k R_{xu}(m-k) \quad (3.4.29)$$

$$R_{xx}(m) = -\sum_{k=1}^{p} a_k R_{xx}(m-k) + \sum_{k=0}^{q} b_k R_{xu}(m-k) \quad (3.4.29)$$

输入 $u(n)$ 与输出 $x(n)$之间的互相关序列为

$$R_{xu}(m) = E[u(n+m)x^*(n)] \quad (3.4.30)$$

将式(3.4.2) $x(n) = \sum_{k=0}^{\infty} h(k)u(n-k)$ 代入上式，可得

$$R_{xu}(m) = E[u(n+m)\sum_{k=0}^{\infty} h^*(k)u^*(n-k)]$$

$$= \sum_{k=0}^{\infty} h^*(k)E[u(n+m)u^*(n-k)]$$

$$= \sum_{k=0}^{\infty} h^*(k)R_{uu}(m+k) \quad (3.4.31)$$

因为 $u(n)$ 为方差为 σ^2 的白噪声，所以下式

$$R_{xu}(m) = \sum_{k=0}^{\infty} h^*(k) R_{uu}(m+k) \qquad (3.4.31)$$

可以化简成

$$R_{xu}(m) = \sigma^2 h^*(-m) \qquad (3.4.32)$$

因为 $h(m)$ 是因果序列，即 $h(-m)=0\ (m>0)$，所以

$$R_{xu}(m) = 0\ (m>0)$$

在条件 $R_{xu}(m)=0\ (m>0)$约束下，式(3.4.29)

$$R_{xx}(m)=-\sum_{k=1}^{p}a_k R_{xx}(m-k)+\sum_{k=0}^{q}b_k R_{xu}(m-k)$$

可化为

$$R_{xx}(m)=\begin{cases}-\sum_{k=1}^{p}a_k R_{xx}(m-k)+\sum_{k=m}^{q}b_k R_{xu}(m-k), & 0\le m\le q\\ -\sum_{k=1}^{p}a_k R_{xx}(m-k), & m\ge q+1\end{cases} \tag{3.4.33}$$

将(3.4.31) $R_{xu}(m)=\sum_{k=0}^{\infty}h^{*}(k)R_{uu}(m+k)$ 代入，上式可得

$$R_{xx}(m)=\begin{cases}-\sum_{k=1}^{p}a_kR_{xx}(m-k)+\sigma^2\sum_{k=m}^{q}b_kh(k-m), & 0\le m\le q\\ -\sum_{k=1}^{p}a_kR_{xx}(m-k), & m\ge q+1\end{cases}$$

(3.4.34)

即

$$R_{xx}(m)=\begin{cases}-\sum_{k=1}^{p}a_kR_{xx}(m-k)+\sigma^2\sum_{k=0}^{q-m}b_{k+m}h(k), & 0\le m\le q\\ -\sum_{k=1}^{p}a_kR_{xx}(m-k), & m\ge q+1\end{cases}$$

(3.4.35)

上式为ARMA(p,q)模型的Yule-Walker方程

MA(q)模型的Yule-Walker方程

将 $a_0 = 1$ 和 $a_k = 0 \ (1 \le k \le p)$ 代入式 (3.4.35)

$$R_{xx}(m) = \begin{cases} -\sum_{k=1}^{p} a_k R_{xx}(m-k) + \sigma^2 \sum_{k=0}^{q-m} b_{k+m} h(k), & 0 \le m \le q \\ -\sum_{k=1}^{p} a_k R_{xx}(m-k), & m \ge q+1 \end{cases}$$

并利用 $H(z) = B(z)$ 或 $h(k) = b_k$，得到

$$R_{xx}(m) = \begin{cases} \sigma^2 \sum_{k=0}^{q-m} b_{k+m} b_k, & 0 \le m \le q \\ 0, & m \ge q+1 \end{cases} \qquad (3.4.36)$$

上式为MA(q)模型的Yule-Walker方程

AR(p)模型的Yule-Walker方程

将 $b_0 = 1$ 和 $b_k = 0\ (1 \le k \le q)$ 代入式 (3.4.35)

$$R_{xx}(m) = \begin{cases} -\sum_{k=1}^{p} a_k R_{xx}(m-k) + \sigma^2 \sum_{k=0}^{q-m} b_{k+m} h(k), & 0 \le m \le q \\ -\sum_{k=1}^{p} a_k R_{xx}(m-k), & m \ge q+1 \end{cases}$$

得到

$$R_{xx}(m) = \begin{cases} -\sum_{k=1}^{p} a_k R_{xx}(m-k) + \sigma^2, & m = 0 \\ -\sum_{k=1}^{p} a_k R_{xx}(m-k), & m \ge 1 \end{cases} \qquad (3.4.37)$$

因为 $a_0 = 1$，下式

$$R_{xx}(m) = \begin{cases} -\sum_{k=1}^{p} a_k R_{xx}(m-k) + \sigma^2, & m = 0 \\ -\sum_{k=1}^{p} a_k R_{xx}(m-k), & m \geq 1 \end{cases} \qquad (3.4.37)$$

可以写成

$$\sum_{k=0}^{p} a_k R_{xx}(m-k) = \begin{cases} \sigma^2, & m = 0 \\ 0, & m \geq 1 \end{cases} \qquad (3.4.38)$$

上式为AR(p)模型的Yule-Walker方程。

AR(p)模型谱估计

用模型方法进行谱估计的一般步骤是：

- ✓ 首先，根据已知观测数据估计自相关函数；
- ✓ 然后，将估计的自相关函数代入Yule-Warlk方程并求解方程；
- ✓ 最后，将解得的模型参数代入计算功率谱的公式。

由于ARMA、和AM模型的Yule-Walk方程为非线性，而AR模型的Yule-Walker方程为线性，便于求解，因此，以下部分讨论AR模型的谱估计问题。

前面已经推导出AR(p)模型的Yule-Walker方程为

$$\sum_{k=0}^{p} a_k R_{xx}(m-k) = \begin{cases} \sigma^2, & m=0 \\ 0, & m \geq 1 \end{cases} \qquad (3.4.39)$$

上述方程包含 $p+1$ 个未知数 $\{a_1, a_2, \cdots, a_p, \sigma^2\}$，将其写成矩阵形式为

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(p) \\ R(1) & R(0) & \cdots & R(p-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(p) & R(p-1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad (3.4.40)$$

根据观测数据 $x(n)$，估计出 $p+1$ 个自相关函数

$$\{R(0), R(1), \cdots, R(p)\}$$

然后用任何一种线性方程组解法求解方程(3.4.40)，得到 $p+1$ 个模型参数 $\{a_1, a_2, \cdots, a_p, \sigma^2\}$ ，最后将模型参数代入式(3.4.14)计算功率谱

$$S_{ar}(e^{j\omega}) = \frac{\sigma^2}{\left|1 + \sum_{k=1}^{p} a_k e^{-j\omega k}\right|^2} \qquad (3.4.41)$$

§3.5 AR谱估计的性质

3.5.3 AR过程的线性预测

- 设$\{x(n)\}$是一个AR(p)过程。它的线性预测，是指用已知观测数据 $x(n-1),x(n-2),\cdots,x(n-p)$ 的线性组合预测未知观测数据 $x(n)$ ，即$x(n)$的预测值为

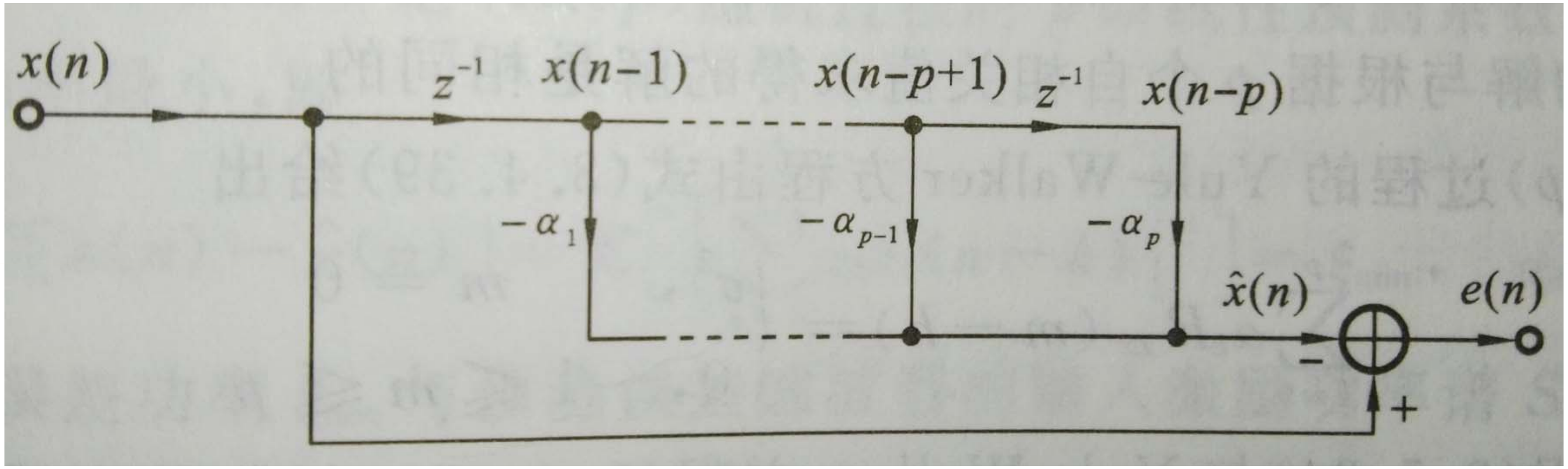
$$\hat{x}(n)=-\sum_{k=1}^{p}\alpha_k x(n-k) \qquad (3.5.26)$$

式中，预测系数按照使预测误差平均(或均方值)最小化准则确定，即

$$\xi=E\left[e^2(n)\right]=\min \qquad (3.5.27)$$

预测误差定义为

$$e(n)=x(n)-\hat{x}(n)=\sum_{k=0}^{p}\alpha_k x(n-k),\ \alpha_0=1 \qquad (3.5.28)$$



注意：由于假设$\{x(n)\}$是广义平稳的，因此，虽然要预测的是$x(n)$，但是预测系数$\alpha_k(k=1,2,\cdots,p)$仅取决于自相关函数，而与时间n无关。

□ 确定预测系数的值

令均方误差 $\xi = E\left[e^2(n)\right]$ 关于$\alpha_j\,(j = 1, 2, \cdots, p)$ 的导数为零，则有 (3.5.29)

$$\frac{\partial \xi(n)}{\partial \alpha_j} = 2E\left[e(n)\frac{\partial e(n)}{\partial \alpha_k}\right] = 2E\left[e(n)\frac{\partial \sum_{j=0}^{p} \alpha_j x(n-k)}{\partial \alpha_j}\right] = 0$$

即 $E\left[e(n)x(n-j)\right] = 0,\ 1 \le j \le p$ (3.5.30)

上式称为线性预测的正交方程。

将误差信号

$$e(n)=x(n)-\hat{x}(n)=\sum_{k=0}^{p}\alpha_k x(n-k),\ \alpha_0=1$$

代入正交方程

$$E\left[e(n)x(n-j)\right]=0,\ 1\le j\le p$$

得到

$$\sum_{k=0}^{p}\alpha_k E\left[x(n-k)x(n-j)\right]=0,\ 1\le j\le p,\alpha_0=1$$

考虑到 $\alpha_0=1$，可以将

$$\sum_{k=0}^{p}\alpha_k E\left[x(n-k)x(n-j)\right]=0,\ 1\le j\le p,\alpha_0=1$$

写成

$$E\left[x(n)x(n-j)\right]=-\sum_{k=1}^{p}\alpha_k E\left[x(n-k)x(n-j)\right],\ 1\le j\le p$$

或者

$$R_x(j)=-\sum_{k=1}^{p}\alpha_k R_x(j-k),\ 1\le j\le p \qquad (3.5.31)$$

其中，

$$R_x(j-k)=E\left[x(n-k)x(n-j)\right],1\le j\le p,0\le k\le p$$

(3.5.32)

当满足正交原理时，式(3.5.27)表示的均方误差达到最小值，此时的最小均方误差为

$$\xi_{\min} = E\left\{e(n)\left[x(n)-\hat{x}(n)\right]\right\} = E\left[e(n)x(n)\right] - E\left[e(n)\hat{x}(n)\right]$$

$$= E\left[e(n)x(n)\right] - \sum_{k=1}^{p}\alpha_k E\left[e(n)x(n-k)\right] = E\left[e(n)x(n)\right]$$

$$= E\left\{\left[\sum_{k=0}^{p}\alpha_k x(n-k)\right]x(n)\right\}$$

$$= E\left[x^2(n)\right] + \sum_{k=1}^{p}\alpha_k E\left[x(n-k)x(n)\right]$$

$$= R_x(0) + \sum_{k=1}^{p}\alpha_k R_x(k) \qquad (3.5.33)$$

将式 $\xi_{\min} = R_x(0) + \sum_{k=1}^{p} \alpha_k R_x(k)$ (3.5.33)

与式 $R_x(j) = -\sum_{k=1}^{p} \alpha_k R_x(j-k),\ 1 \le j \le p$ (3.5.31)

合并，得到

$$R_x(j) = \begin{cases} \xi_{\min} - \sum_{k=1}^{p} \alpha_k R_x(k), & j = 0 \\ -\sum_{k=1}^{p} \alpha_k R_x(j-k), & 1 \le j \le p \end{cases}$$

因为 $\alpha_0=1$，且 $R_x(k)=R_x(-k)$，下式

$$R_x(j)=\begin{cases}\xi_{\min}-\sum_{k=1}^{p}\alpha_k R_x(k), & j=0\\ -\sum_{k=1}^{p}\alpha_k R_x(j-k), & 1\le j\le p\end{cases}$$

可写成

$$\sum_{k=0}^{p}\alpha_k R_x(j-k)=\begin{cases}\xi_{\min}, & j=0\\ 0, & 1\le j\le p\end{cases}\qquad(3.5.34)$$

上式称为**线性预测的Wiener-Hopf方程**。

前面已经推导出，AR(p)过程的Yule-Walker方程由下式给出

$$\sum_{k=0}^{p} a_k R_{xx}(m-k) = \begin{cases} \sigma^2, & m=0 \\ 0, & 1 \le m \le p \end{cases} \qquad (3.5.35)$$

将其与如下的线性预测的Wiener-Hopf方程相比较

$$\sum_{k=0}^{p} \alpha_k R_x(j-k) = \begin{cases} \xi_{\min}, & j=0 \\ 0, & 1 \le j \le p \end{cases} \qquad (3.5.34)$$

可见两者的结构相同。

如果两个自相关矩阵 $R_x(j-k)\ (0 \le j,k \le p)$ 和 $R_x(m-k)\ (0 \le m,k \le p)$ 都是正定的，则线性预测的**Wiener-Hopf方程**和**Yule-Walker方程**分别有唯一解

$$\{\alpha_1,\alpha_2,\cdots,\alpha_p,\xi_{\min}\} \quad 和 \quad \{a_1,a_2,\cdots,a_p,\sigma^2\}$$

如果这两个自相关矩阵完全相同，则两个方程的解也相同，即

$$\{\alpha_1,\alpha_2,\cdots,\alpha_p,\xi_{\min}\}=\{a_1,a_2,\cdots,a_p,\sigma^2\} \qquad (3.5.36)$$

解也相同 $\{\alpha_1,\alpha_2,\cdots,\alpha_p,\xi_{\min}\}=\{a_1,a_2,\cdots,a_p,\sigma^2\}$

也就是说，AR过程的线性预测系数正是AR模型参数，最小预测均方误差正好等于AR模型的输入激励白噪声的方差。

只有两者阶数相同的情况下，上述结论才是正确的。在此情况下，由式(3.5.36)可得

$$e(n)=x(n)+\sum_{k=1}^{p}\alpha_k x(n-k)=x(n)+\sum_{k=1}^{p}a_k x(n-k)=u(n) \tag{3.5.37}$$

§3.6 Levinson-Durbin算法

3.6.1 Levinson-Durbin算法的推导

- 前面建立的Yule-Walker方程

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(p) \\ R(1) & R(0) & \cdots & R(p-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(p) & R(p-1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

如果采用线性方程组的常用解法，如高斯消元法，需要的运算量数量级为$O(p^3)$，当 p 很大时，计算量很大。

Levinson-Durbin算法是一种阶递归运算的算法:

- ✓ 以AR(0)模型参数为起始条件计算AR(1)模型参数;
- ✓ 利用AR(1)模型参数计算AR(2)模型参数;
- ✓ 利用AR(2)模型参数计算AR(3)模型参数;
- ✓ ………………
- ✓ 利用AR(p-1)模型参数计算AR(p)模型参数。

Levinson-Durbin算法的价值:

- ✓ 把运算量减少为$O(p^2)$数量级;
- ✓ 得到了所有等于和低于p阶的所有各阶模型参数;
- ✓ 导出了线性预测的反射系数和格型滤波器结构。

k阶和k+1阶的Yule-Warlk方程分别为

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) \\ R(1) & R(0) & \cdots & R(k-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \end{bmatrix} = -\begin{bmatrix} \sigma_k^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad (3.6.1)$$

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{k+1,1} \\ \vdots \\ a_{k+1,k} \\ a_{k+1,k+1} \end{bmatrix} = \begin{bmatrix} \sigma_{k+1}^2 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \quad (3.6.2)$$

$a_{i,j}$ 的第一个下标表示模型阶数，第二个下标表示序号。

在列向量 $[1, a_{k,1}, \cdots, a_{k,k}]^T$ 后增加一个零元素，并将其与 $k+1$ 阶系数矩阵相乘，可得

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} = \begin{bmatrix} \sigma_k^2 \\ 0 \\ \vdots \\ 0 \\ D_k \end{bmatrix} \qquad (3.6.3)$$

上式称为**扩展方程**，它比阶Yule-Walker方程增加了一个方程

$$D_k = R(k+1) + \sum_{i=1}^{k} a_{ki} R(k+1-i) \qquad (3.6.4)$$

因为如下扩展方程的系数矩阵满足Toeplitz性质

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} = \begin{bmatrix} \sigma_k^2 \\ 0 \\ \vdots \\ 0 \\ D_k \end{bmatrix} \quad (3.6.3)$$

则可将方程(3.6.3)写成倒序方程

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 0 \\ a_{k,k} \\ \vdots \\ a_{k,1} \\ 1 \end{bmatrix} = \begin{bmatrix} D_k \\ 0 \\ \vdots \\ 0 \\ \sigma_k^2 \end{bmatrix} \quad (3.6.5)$$

扩展方程、倒序方程和待求解的$k+1$阶Yule-Walker方程具有相同的系数矩阵。将扩展方程和倒序方程进行线性组合，得到如下新的方程

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \left\{ \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} - \gamma_{k+1} \begin{bmatrix} 0 \\ a_{k,k} \\ \vdots \\ a_{k,1} \\ 1 \end{bmatrix} \right\} = \begin{bmatrix} \sigma_k^2 \\ 0 \\ \vdots \\ 0 \\ D_k \end{bmatrix} - \gamma_{k+1} \begin{bmatrix} D_k \\ 0 \\ \vdots \\ 0 \\ \sigma_k^2 \end{bmatrix}$$

(3.6.6)

如果选择加权参数 γ_{k+1} 使下式成立

$$\gamma_{k+1}=\frac{D_k}{\sigma_k^2} \qquad (3.6.7)$$

那么，组合式(3.6.7)变成

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) \cdots & R(0) \end{bmatrix} \left\{ \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} - \gamma_{k+1} \begin{bmatrix} 0 \\ a_{k,k} \\ \vdots \\ a_{k,1} \\ 1 \end{bmatrix} \right\} = \begin{bmatrix} \sigma_k^2(1-\gamma_{k+1}) \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

$$(3.6.8)$$

比较$k+1$阶组合方程和$k+1$阶Yule-Walker方程

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \left\{ \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} - \gamma_{k+1} \begin{bmatrix} 0 \\ a_{k,k} \\ \vdots \\ a_{k,1} \\ 1 \end{bmatrix} \right\} = \begin{bmatrix} \sigma_k^2(1-\gamma_{k+1}) \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(k) & R(k+1) \\ R(1) & R(0) & \cdots & R(k-1) & R(k) \\ \vdots & \vdots & \ddots & \vdots & \cdots & \vdots \\ R(k) & R(k-1) & \cdots & R(0) & \cdots & R(1) \\ R(k+1) & R(k) & \cdots & R(1) & \cdots & R(0) \end{bmatrix} \begin{bmatrix} 1 \\ a_{k+1,1} \\ \vdots \\ a_{k+1,k} \\ a_{k+1,k+1} \end{bmatrix} = \begin{bmatrix} \sigma_{k+1}^2 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

可见，如果令右式相等，则解也相等。

即

$$\begin{bmatrix} 1 \\ a_{k+1,1} \\ \vdots \\ a_{k+1,k} \\ a_{k+1,k+1} \end{bmatrix} = \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} - \gamma_{k+1} \begin{bmatrix} 0 \\ a_{k,k} \\ \vdots \\ a_{k,1} \\ 1 \end{bmatrix} \qquad (3.6.9)$$

$$\sigma_{k+1}^2 = \sigma_k^2 (1 - \gamma_{k+1}^2) \qquad (3.6.10)$$

上两式为已知k阶Yule-Walker方程的已知解

$$\{a_{k,1}, \cdots, a_{k,k}, \sigma_k^2\}$$

求$k+1$阶Yule-Walker方程的未知解

$$\{a_{k+1,1}, \cdots, a_{k+1,k+1}, \sigma_{k+1}^2\}$$

的递推公式。

式(3.6.9) $\begin{bmatrix} 1 \\ a_{k+1,1} \\ \vdots \\ a_{k+1,k} \\ a_{k+1,k+1} \end{bmatrix} = \begin{bmatrix} 1 \\ a_{k,1} \\ \vdots \\ a_{k,k} \\ 0 \end{bmatrix} - \gamma_{k+1} \begin{bmatrix} 0 \\ a_{k,k} \\ \vdots \\ a_{k,1} \\ 1 \end{bmatrix}$ 的紧凑形式为

$$a_{k+1,0} = a_{k,0} = 1, \quad a_{k,k+1} = 0$$

$$a_{k+1,i} = a_{k,i} - \gamma_{k+1} a_{k,k+1-i}\,, \quad i = 1, 2, \cdots, k+1 \qquad (3.6.11)$$

从式(3.6.9)的最后一行，可以看出

$$a_{k+1,k+1} = -\gamma_{k+1} \qquad (3.6.12)$$

γ_{k+1} 称为反射系数，可以直接用式(3.6.7) $\gamma_{k+1} = \dfrac{D_k}{\sigma_k^2}$ 计算。

将式(3.6.4) $D_k = R(k+1) + \sum_{i=1}^{k} a_{ki} R(k+1-i)$

代入式(3.6.7) $\gamma_{k+1} = \dfrac{D_k}{\sigma_k^2}$，得到

$$\gamma_{k+1} = \frac{R(k+1) + \sum_{i=1}^{k} a_{k,i} R(k+1-i)}{\sigma_k^2} \qquad (3.6.13)$$

$$\sigma_{k+1}^2 = \sigma_k^2 (1 - \gamma_{k+1}^2) \qquad (3.6.10)$$

上式中的 σ_k^2，可以由(3.6.1)的第一个方程求出：

$$\sigma_k^2 = R(0) + \sum_{i=1}^{k} a_{k,i} R(i) \qquad (3.6.14)$$

Levinson-Durbin算法的迭代计算步骤归纳如下：

(1)初始化：$a_{0,0} = a_{1,0} \cdots a_{p,0} = 1;\ \sigma_0^2 = R(0)$

(2)迭代运算 $(k = 0,1,\cdots,p-1)$

计算阶反射系数 γ_{k+1}

$$\gamma_{k+1} = \frac{R(k+1) + \sum_{i=1}^{k} a_{k,i} R(k+1-i)}{\sigma_k^2}$$

计算 $a_{k+1,i}\ (i = 1,2,\cdots,k)$

$$a_{k+1,i} = a_{k,i} - \gamma_{k+1} a_{k,k+1-i},\quad a_{k+1,k+1} = -\gamma_{k+1}$$

更新激励白噪声方程

$$\sigma_{k+1}^2 = (1-\gamma_{k+1}^2)\sigma_k^2$$

Levinson-Durbin算法，实际上是把信号的自相关序列

$$\{R(0), R(1), \cdots, R(p)\}$$

映射成AR模型参数

$$\{a_{p,k}; k = 1,2,\cdots, p\} \quad 和 \quad \{\sigma_k^2; k = 1,2,\cdots, p\}$$

同时还附带产生一组反射系数

$$\{\lambda_k; k = 1,2,\cdots, p\}$$

3.6.2 格型滤波器

格型滤波器结构可以由正向和反向预测误差的递推计算公式导出。

$$\hat{x}(n)=-\sum_{i=1}^{k}a_{k,i}x(n-i)$$

正向预测误差定义为

$$e_k^f(n)=x(n)-\hat{x}(n)=x(n)+\sum_{i=1}^{k}a_{k,i}x(n-i) \qquad (3.6.16)$$

式中，上标 f 表示正向预测，下标 k 表示线性预测的阶。上式表示， k 阶正向预测是指用信号在 n 时刻以前的 k 个取样值$x(n-1),x(n-2),\cdots,x(n-k)$预测 $x(n)$ 。

由式(3.6.16)

$$e_k^f(n) = x(n) - \hat{x}(n) = x(n) + \sum_{i=1}^{k} a_{k,i} x(n-i) \qquad (3.6.16)$$

可写出k阶正向预测误差滤波器的传递函数

$$A_k(z) = \frac{E_k^f(z)}{X(z)} = 1 + \sum_{i=1}^{k} a_{k,i} z^{-i} \qquad (3.6.17)$$

式中，$E_k^f(z)$是$e_k^f(n)$的 z 变换。

k 阶反向预测误差定义为 $\hat{x}(n) = -\sum_{i=1}^{k} a_{k,i} x(n-k+i)$

$$e_k^b(n) = x(n-k) - \hat{x}(n-k)$$

$$= x(n-k) + \sum_{i=1}^{k} a_{k,i} x(n-k+i) \quad (3.6.18)$$

$$= x(n-k) + \sum_{i=0}^{k-1} a_{k,k-i} x(n-i) \quad (3.6.19)$$

式中，上标 b 表示反向预测，下标 k 表示线性预测的阶。上式表示，k 阶反向预测是指用信号在 n-k 时刻以后的 k 个取样值

$$x(n), x(n-1), \cdots, x(n-k+1) \overset{预测}{\Rightarrow} x(n-N)$$

由式(3.6.19)

$$e_k^b(n) = x(n-k) + \sum_{i=1}^{k} a_{k,i} x(n-k+i) \qquad (3.6.19)$$

可写出k阶反向预测误差滤波器的传递函数

$$A_k^R(z) = \frac{E_k^b(z)}{X(z)} = z^{-k} + \sum_{i=1}^{k} a_{k,i} z^{-(k-i)} = z^{-k}\left[1 + \sum_{i=1}^{k} a_{k,i} z^{i}\right]$$

$$= z^{-k} A_k(z^{-1}) \qquad (3.6.20)$$

$$A_k(z) = 1 + \sum_{i=1}^{k} a_{k,i} z^{-i}$$

式中，$E_k^b(z)$是$e_k^b(n)$的 z变换。

可以看出，$A_k^R(z)$是$A_k(z^{-1})$的倒序多项式。

对Levinson-Durbin算法的预测系数递推计算公式

$$a_{k+1,i} = a_{k,i} - \gamma_{k+1} a_{k,k+1-i}\ ,\quad i = 1,2,\cdots,k+1 \qquad (3.6.11)$$

取z变换，得到

$$A_{k+1}(z) = A_k(z) - \gamma_{k+1} z^{-1} A_k^R(z) \qquad (3.6.21)$$

提示：$a_{k,i} \leftrightarrow A_k(z) \Rightarrow a_{k,-i} \leftrightarrow A_k(z^{-1})$

$$\Rightarrow a_{k,k+1-i} = a_{k,-(i-(k+1))} \leftrightarrow z^{-(k+1)} A_k(z^{-1}) = z^{-1} \underbrace{z^{-k} A_k(z^{-1})}_{A_k^R(z)}$$

把(3.6.21)倒序，得到它的等效表示式

$$A_{k+1}^R(z) = z^{-1} A_k^R(z) - \gamma_{k+1} A_k(z) \qquad (3.6.22)$$

将 $\begin{cases} A_{k+1}(z) = A_k(z) - \gamma_{k+1} z^{-1} A_k^R(z) \\ A_{k+1}^R(z) = -\gamma_{k+1} A_k(z) + z^{-1} A_k^R(z) \end{cases}$ 写成矩阵形式，可得

$$\begin{bmatrix} A_{k+1}(z) \\ A_{k+1}^R(z) \end{bmatrix} = \begin{bmatrix} 1 & -\gamma_{k+1} z^{-1} \\ -\gamma_{k+1} & z^{-1} \end{bmatrix} \begin{bmatrix} A_k(z) \\ A_k^R(z) \end{bmatrix} \quad (3.6.23)$$

将上式左右两端同乘 $X(z)$，并利用

$$A_k(z) = \frac{E_k^f(z)}{X(z)} \quad (3.6.17) \qquad A_k^R(z) = \frac{E_k^b(z)}{X(z)} \quad (3.6.20)$$

可得

$$\begin{bmatrix} E_{k+1}^f(z) \\ E_{k+1}^b(z) \end{bmatrix} = \begin{bmatrix} 1 & -\gamma_{k+1} z^{-1} \\ -\gamma_{k+1} & z^{-1} \end{bmatrix} \begin{bmatrix} E_k^f(z) \\ E_k^b(z) \end{bmatrix} \quad (3.6.24)$$

对式(3.6.24)

$$\begin{bmatrix} E_{k+1}^{f}(z) \\ E_{k+1}^{b}(z) \end{bmatrix} = \begin{bmatrix} 1 & -\gamma_{k+1} z^{-1} \\ -\gamma_{k+1} & z^{-1} \end{bmatrix} \begin{bmatrix} E_{k}^{f}(z) \\ E_{k}^{b}(z) \end{bmatrix} \quad (3.6.24)$$

取逆 Z 变换可得

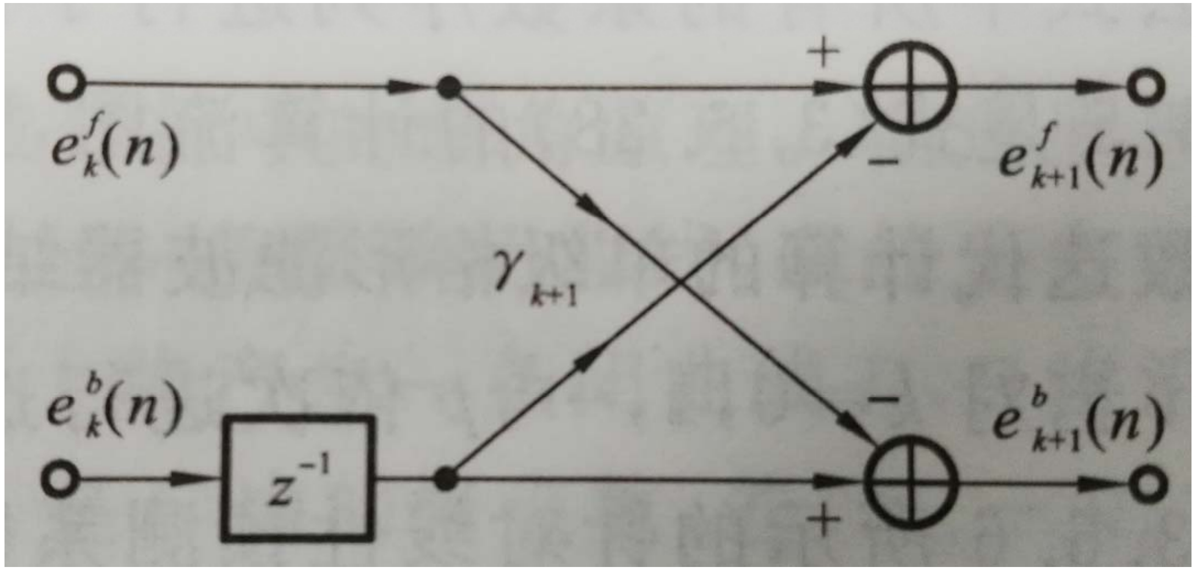
$$\begin{bmatrix} e_{k+1}^{f}(n) \\ e_{k+1}^{b}(n) \end{bmatrix} = \begin{bmatrix} 1 & -\gamma_{k+1} \\ -\gamma_{k+1} & 1 \end{bmatrix} \begin{bmatrix} e_{k}^{f}(n) \\ e_{k}^{b}(n-1) \end{bmatrix} \quad (3.6.25)$$

上式为正向预测误差和反向预测误差的递推计算公式，它定义了单级分析格型滤波器结构。

递推式(3.6.25)

$$\begin{bmatrix} e_{k+1}^f(n) \\ e_{k+1}^b(n) \end{bmatrix} = \begin{bmatrix} 1 & -\gamma_{k+1} \\ -\gamma_{k+1} & 1 \end{bmatrix} \begin{bmatrix} e_k^f(n) \\ e_k^b(n-1) \end{bmatrix} \qquad (3.6.25)$$

定义的单级分析格型滤波器结构如下图所示



根据

$$A_k(z)=\frac{E_k^f(z)}{X(z)}=1+\sum_{i=1}^{k}a_i z^{-i} \qquad (3.6.17)$$

$$A_k^R(z)=\frac{E_k^b(z)}{X(z)}=z^{-k}+\sum_{i=1}^{k}a_{k,i}z^{-(k-i)} \qquad (3.6.20)$$

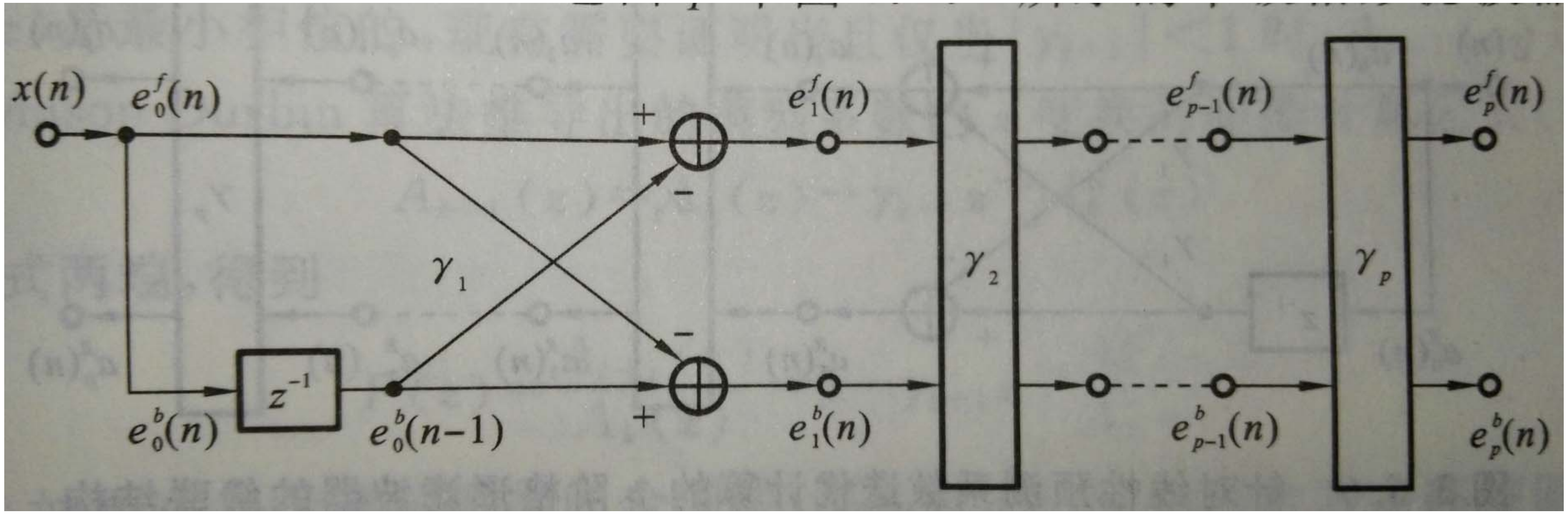
当 $k=0$ 时，有 $A_0(z)=1,\ A_0^R(z)=1$，

因此

$$E_0^f(z)=E_0^b(z)=X(z)$$

$$e_0^f(n)=e_0^b(n)=x(n)$$

因此，p阶**分析格型滤波器**结构如下图所示



其由p个**单级分析格型滤波器**组成。

§3.7 根据有限长观测数据序列估计AR(p)模型参数

实际应用中：

- 只能观测到随机过程的一个取样序列的一段(有限个)取样值；
- 只能根据有限个观测数据估计AR(p)模型参数。

常用的方法有：

- 自相关法；
- 协方差法；
- 修正协方差法；
- Burg法。

上述方法都优于经典谱法，且有相应的快速算法

设随机过程的观测数据为

$$x(n) = \{x(0), x(1), \cdots, x(N-1)\}$$

p阶预测误差为

$$e_p(n) = x(n) - \hat{x}_p(n) = \sum_{j=0}^{p} a_{p,j} x(n-j), \; a_{p,0} = 1 \quad (3.7.1)$$

其中，$\hat{x}_p(n)$为p阶预测值

$$\hat{x}_p(n) = -\sum_{j=1}^{p} a_{p,j} x(n-j) \qquad (3.7.2)$$

式(3.7.1) $e_p(n)=x(n)-\hat{x}_p(n)=\sum_{j=0}^{p}a_{p,j}x(n-j),\ a_{p,0}=1$

的展开形式为

$$\begin{bmatrix} e_p(0) \\ \vdots \\ e_p(p) \\ \vdots \\ e_p(N-1) \\ \vdots \\ e_p(N-1+p) \end{bmatrix} = \begin{bmatrix} x(0) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ x(p) & \cdots & x(0) \\ \vdots & & \vdots \\ x(N-1) & \cdots & x(N-1-p) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & x(N-1) \end{bmatrix} \begin{bmatrix} 1 \\ a_{p,1} \\ \vdots \\ a_{p,p} \end{bmatrix} \quad (3.7.3)$$

由上式可见，由于 $x(n)$ 为有限长，

所以 $e_p(n)(0\le n\le N-1+p)$ 也为有限长。

3.7.4 Burg法

$$\begin{bmatrix} e_p(0) \\ \vdots \\ e_p(p) \\ \vdots \\ e_p(N-1) \\ \vdots \\ e_p(N-1+p) \end{bmatrix} = \begin{bmatrix} x(0) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ x(p) & \cdots & x(0) \\ \vdots & & \vdots \\ x(N-1) & \cdots & x(N-1-p) \\ \vdots & \ddots & \vdots \\ 0 & \cdots & x(N-1) \end{bmatrix} \begin{bmatrix} 1 \\ a_{p,1} \\ \vdots \\ a_{p,p} \end{bmatrix} \quad (3.7.3)$$

定义代价函数 $\xi_k = \sum_{n=k}^{N-1} \left\{ \left[e_k^f(n) \right]^2 + \left[e_k^b(n) \right]^2 \right\}$ (3.7.54)

根据式 $\begin{bmatrix} e_{k+1}^{f}(n) \\ e_{k+1}^{b}(n) \end{bmatrix} = \begin{bmatrix} 1 & -\gamma_{k+1} \\ -\gamma_{k+1} & 1 \end{bmatrix} \begin{bmatrix} e_{k}^{f}(n) \\ e_{k}^{b}(n-1) \end{bmatrix}$ (3.6.25)

有如下递推关系式

$$e_k^f(n) = e_{k-1}^f(n) - \gamma_k e_{k-1}^b(n-1) \qquad (3.7.55)$$

$$e_k^b(n) = -\gamma_k e_{k-1}^f(n) + e_{k-1}^b(n-1) \qquad (3.7.56)$$

令 $\xi_k = \sum_{n=k}^{N-1} \left\{ \left[e_k^f(n) \right]^2 + \left[e_k^{\mathrm{b}}(n) \right]^2 \right\}$ 对 γ_k 的偏导数

等于零，并采用以上两式，可得

$$\frac{\partial \xi_k}{\partial \gamma_k} = \sum_{n=k}^{N-1}\left[e_k^f(n)e_{k-1}^b(n-1) + e_k^b(n)e_{k-1}^f(n)\right] = 0 \quad (3.7.57)$$

将以下两式

$$e_k^f(n) = e_{k-1}^f(n) - \gamma_k e_{k-1}^b(n-1) \quad (3.7.55)$$

$$e_k^b(n) = -\gamma_k e_{k-1}^f(n) + e_{k-1}^b(n-1) \quad (3.7.56)$$

代入(3.7.57)，可解得 γ_k ，即

$$\gamma_k = \frac{2\sum_{n=k}^{N-1} e_{k-1}^f(n)e_{k-1}^b(n-1)}{\sum_{n=k}^{N-1}\left\{\left[e_{k-1}^f(n)\right]^2 + \left[e_{k-1}^{\mathrm{b}}(n-1)\right]^2\right\}} \quad (3.7.58)$$

将式 $$\gamma_k=\frac{2\sum_{n=k}^{N-1}e_{k-1}^{f}(n)e_{k-1}^{b}(n-1)}{\sum_{n=k}^{N-1}\left\{\left[e_{k-1}^{f}(n)\right]^2+\left[e_{k-1}^{\mathrm{b}}(n-1)\right]^2\right\}} \qquad (3.7.58)$$

写成向量形式如下向量形式

$$\gamma_k=\frac{2\left\langle \mathbf{e}_{k-1}^{f},\mathbf{e}_{k-1}^{b}\right\rangle}{\left\|\mathbf{e}_{k-1}^{f}\right\|^2+\left\|\mathbf{e}_{k-1}^{b}\right\|^2} \qquad (3.7.59)$$

其中

$$\mathbf{e}_{k-1}^{f}=\left[e_{k-1}^{f}(k),e_{k-1}^{f}(k+1),\cdots,e_{k-1}^{f}(N-1)\right]^{\mathrm{T}} \quad (3.7.60)$$

$$\mathbf{e}_{k-1}^{b}=\left[e_{k-1}^{b}(k-1),e_{k-1}^{b}(k),\cdots,e_{k-1}^{b}(N-2)\right]^{\mathrm{T}} \quad (3.7.61)$$

分别为**前向误测误差向量**和**后向误测误差向量**。

根据Cauchy-Schwartz不等式，有

$$2\left|\left\langle \mathbf{e}_{k-1}^{f}, \mathbf{e}_{k-1}^{b} \right\rangle\right| \leq \left\|\mathbf{e}_{k-1}^{f}\right\|^{2} + \left\|\mathbf{e}_{k-1}^{b}\right\|^{2} \qquad (3.7.62)$$

因此，根据式(3.7.59)

$$\gamma_k = \frac{2\left\langle \mathbf{e}_{k-1}^{f}, \mathbf{e}_{k-1}^{b} \right\rangle}{\left\|\mathbf{e}_{k-1}^{f}\right\|^{2} + \left\|\mathbf{e}_{k-1}^{b}\right\|^{2}} \qquad (3.7.59)$$

可得出 $$\left|\gamma_k\right| \leq 1 \qquad (3.7.61)$$

将式

$$e_k^f(n) = e_{k-1}^f(n) - \gamma_k e_{k-1}^b(n-1) \qquad (3.7.55)$$

$$e_k^b(n) = -\gamma_k e_{k-1}^f(n) + e_{k-1}^b(n-1) \qquad (3.7.56)$$

代入式

$$\xi_k = \sum_{n=k}^{N-1} \left\{ \left[e_k^f(n) \right]^2 + \left[e_k^b(n) \right]^2 \right\} \qquad (3.7.54)$$

可得

$$\xi_k = \sum_{n=k}^{N-1} \left[e_{k-1}^f(n) - \gamma_k e_{k-1}^b(n-1) \right]^2 + \sum_{n=k}^{N-1} \left[-\gamma_k e_{k-1}^f(n) + e_{k-1}^b(n-1) \right]^2$$

$$= [1+\gamma_k^2] \sum_{n=k}^{N-1} \left\{ \left[e_{k-1}^f(n) \right]^2 + \left[e_{k-1}^b(n-1) \right]^2 \right\} - 4\gamma_k \sum_{n=k}^{N-1} \left[e_{k-1}^f(n) e_{k-1}^b(n-1) \right]$$

$$(3.7.64)$$

式 $$\gamma_k=\frac{2\sum_{n=k}^{N-1}e_{k-1}^{f}(n)e_{k-1}^{b}(n-1)}{\sum_{n=k}^{N-1}\left\{\left[e_{k-1}^{f}(n)\right]^2+\left[e_{k-1}^{\mathrm{b}}(n-1)\right]^2\right\}}$$ (3.7.58) 可改写成

$$2\sum_{n=k}^{N-1}e_{k-1}^{f}(n)e_{k-1}^{b}(n-1)=\gamma_k\sum_{n=k}^{N-1}\left\{\left[e_{k-1}^{f}(n)\right]^2+\left[e_{k-1}^{b}(n-1)\right]^2\right\}$$

(3.7.65)

因而式(3.7.64)

$$\xi_k=\sum_{n=k}^{N-1}\left[e_k^{f}(n)-\gamma_k e_{k-1}^{b}(n-1)\right]^2+\sum_{n=k}^{N-1}\left[-\gamma_k e_{k-1}^{f}(n)+e_{k-1}^{b}(n-1)\right]^2$$

$$=[1+\gamma_k^2]\sum_{n=k}^{N-1}\left\{\left[e_{k-1}^{f}(n)\right]^2+\left[e_{k-1}^{b}(n-1)\right]^2\right\}-4\gamma_k\sum_{n=k}^{N-1}\left[e_{k-1}^{f}(n)e_{k-1}^{b}(n-1)\right]$$

可简化成

$$\xi_k=[1-\gamma_k^2]\sum_{n=k}^{N-1}\left\{\left[e_{k-1}^f(n)\right]^2+\left[e_{k-1}^b(n-1)\right]^2\right\}$$

$$=[1-\gamma_k^2]\left\{\sum_{n=k-1}^{N-1}\left(\left[e_{k-1}^f(n)\right]^2+\left[e_{k-1}^b(n)\right]^2\right)-\left[e_{k-1}^f(k-1)\right]^2-\left[e_{k-1}^b(N-1)\right]^2\right\}$$

$$=[1-\gamma_k^2]\left\{\xi_{k-1}-\left[e_{k-1}^f(k-1)\right]^2-\left[e_{k-1}^b(N-1)\right]^2\right\} \qquad (3.7.66)$$

迭代式的初始条件为

$$\xi_0=\sum_{n=0}^{N-1}\left\{\left[e_0^f(n)^2+e_0^b(n)^2\right]\right\}=2\sum_{n=0}^{N-1}x^2(n) \qquad (3.7.67)$$

□下面讨论如何降低式(3.7.59)的计算量

$$\gamma_k = \frac{2\langle \mathbf{e}_{k-1}^f, \mathbf{e}_{k-1}^b \rangle}{\|\mathbf{e}_{k-1}^f\|^2 + \|\mathbf{e}_{k-1}^b\|^2} \qquad (3.7.59)$$

用分母表示D_k，可得

$$D_k = \sum_{n=k}^{N-1} \left\{ \left[e_{k-1}^f(n) \right]^2 + \left[e_{k-1}^b(n-1) \right]^2 \right\} \qquad (3.7.68)$$

上式的等效形式为

$$D_k = \sum_{n=k-1}^{N-1} \left\{ \left[e_{k-1}^f(n) \right]^2 + \left[e_{k-1}^b(n-1) \right]^2 \right\} - e_{k-1}^f(k-1) - e_{k-1}^b(N-1)$$

$$= \xi_{k-1} - \left[e_{k-1}^f(k-1) \right]^2 - \left[e_{k-1}^b(N-1) \right]^2 \qquad (3.7.69)$$

将式(3.7.69) $D_k = \xi_{k-1} - \left[e_{k-1}^{f}(k-1)\right]^2 - \left[e_{k-1}^{b}(N-1)\right]^2$

代入式(3.7.66) $\xi_k = [1-\gamma_k^2]\left\{\xi_{k-1} - \left[e_{k-1}^{f}(k-1)\right]^2 - \left[e_{k-1}^{b}(N-1)\right]^2\right\}$

可得 $\xi_k = (1-\gamma_k^2)D_k$ (3.7.70)

由式(3.7.69)可写出

$$D_{k+1} = \xi_k - \left[e_k^{f}(k)\right]^2 - \left[e_k^{b}(N-1)\right]^2 \quad (3.7.71)$$

将式(3.7.70)代入式(3.7.71)，可得

$$D_{k+1} = (1-\gamma_k^2)D_k - \left[e_k^{f}(k)\right]^2 - \left[e_k^{b}(N-1)\right]^2 \quad (3.7.72)$$